



Mobiliser toutes les connaissances vues jusqu'à présent en NSI sur la programmation en Python.  
Travailler en équipe, avoir une démarche de projet.



Il est nécessaire avant de commencer ce projet de bien avoir pris connaissance de la fiche sur « *Notion de modularité.pdf* » et des fiches relatives au module Tuttle : « *NSI - Module Turtle.pdf* » et « *TurtleRef.pdf* ».

## 1 PRESENTATION DU PROJET

On souhaite écrire un programme qui permet de générer « aléatoirement » le dessin d'une rue de 4 immeubles.  
On utilisera pour cela le module Turtle de Python.

Les contraintes urbanistiques sont les suivantes :

- Les **immeubles** ont :
  - au minimum un rez-de-chaussée et au maximum 4 étages (5 niveaux) ;
  - une largeur de 140 pixels ;
  - une couleur unique pour toute la façade.
- Chaque **niveau** (rez-de-chaussée ou étage) :
  - a une façade de hauteur 60 pixels ;
  - comprend trois éléments également espacés.
- Les **rez-de-chaussée** n'ont qu'une seule porte et 2 fenêtres placées aléatoirement ;
- Une **porte** a une largeur de 30 pixels et une hauteur de 50 pixels, chaque porte ayant une couleur différente, choisie aléatoirement.
- Toutes les **fenêtres** sont identiques, de taille 30 pixels sur 30 pixels, fond bleu clair.
- Toutes les **portes fenêtre** ont un balcon et font une taille de 30 pixels en largeur et 50 pixels en hauteurs, fond bleu clair.
- Le **toit** peut avoir 2 formes : plat ou triangulaire.
  - Si le toit est plat, il fait une épaisseur de 10 pixels.
  - Si le toit est triangulaire, il fait une hauteur de 40 pixels pour une base de 160 pixels.

## 2 EXEMPLES

La série d'exemples ci-contre est basée sur :

- Une couleur aléatoire pour les façades ;
- Deux modèles de toit ;
- Deux modèles (on se limitera à un seul) de porte avec une couleur aléatoire ;
- Deux modèles d'ouvertures pour les étages : fenêtre ou porte-fenêtre avec balcon ;
- Trois éléments horizontalement pour chaque niveau.

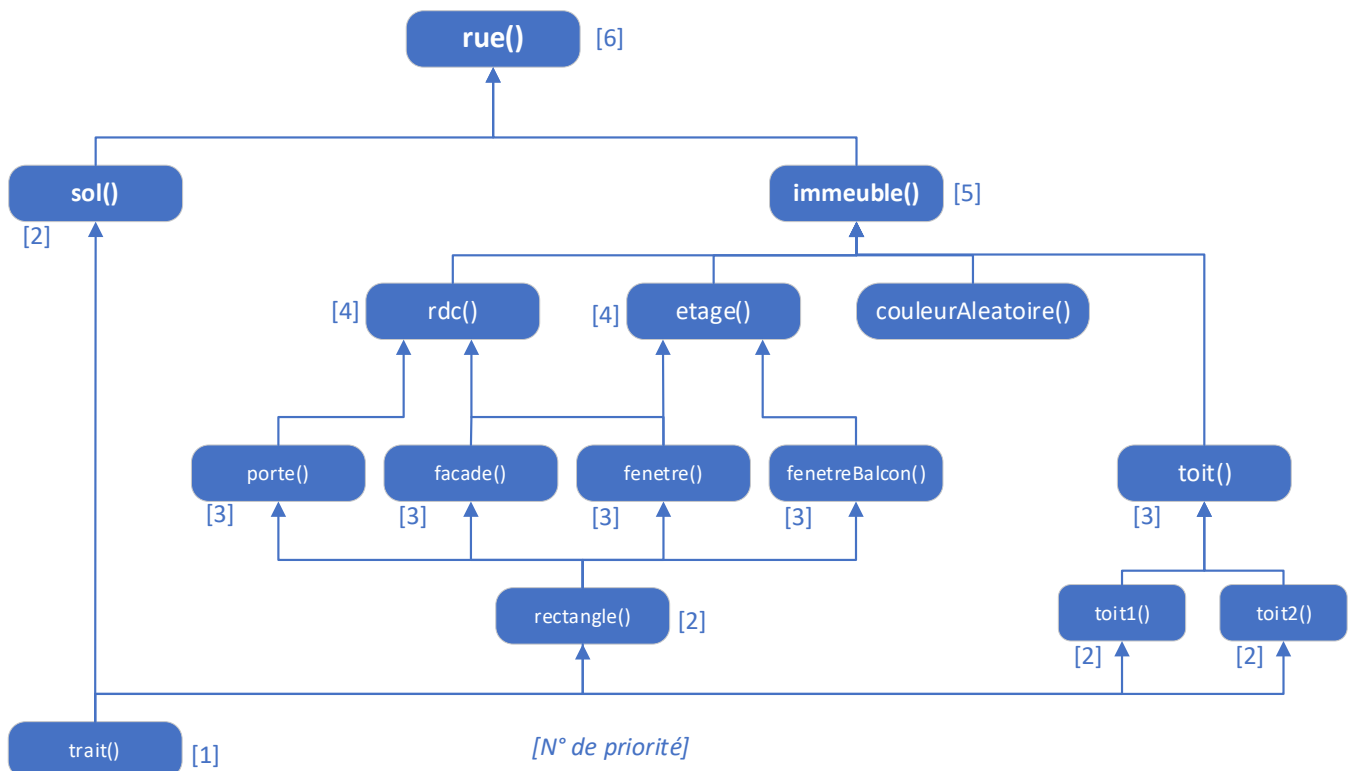


## 3 LE TRAVAIL A FAIRE

Vous devez écrire le programme sous forme de modules qui réponde au problème posé en utilisant le module Turtle de Python (voir le memento joint).

Le programme est réparti en plusieurs modules comportant une seule fonction (du même nom que le module), vous permettant ainsi de travailler seul sur l'un d'entre eux. Mais vous travaillerez collectivement et en interdépendance au travers des importations de modules.

Pour cela toutes les fonctions à écrire ont été recensées. Leurs dépendances sont décrites dans le schéma ci-contre.



Par exemple le module **fenetre** pour sa fonction **fenetre()** aura besoin du module **rectangle** (pour sa fonction éponyme) qui lui-même aura besoin du module **trait**. D'ailleurs ce dernier module est nécessaire pour tous les autres. Il faudra donc commencer par écrire les modules du bas et terminer par le module **rue** qui sera le programme principal.

Pour vous aider dans votre tâche, dans chaque module, les importations et les *docstrings* ont déjà été faits ainsi que les commentaires. Il ne vous reste qu'à écrire le code (remplacer l'instruction *pass* par votre code) en ayant bien compris ce que votre module doit importer avant de commencer. Une image pour chaque module est donnée pour illustrer l'attendu du module à écrire. Attention, il ne faut surtout pas réécrire les fonctions importées, mais bien les utiliser au sein de votre module.

La structure type d'un module est le suivant :

### Exemple du module « etage.py »

```
from facade import facade
from random import shuffle, randint
from fenetre import fenetre
from fenetre_balcon import fenetre_balcon
import turtle

def etage(x, y_sol, couleur, niveau):
    '''
    Paramètres
    x : abscisse du centre de L'étage
    y_sol : ordonnée du sol de la rue
    couleur : couleur de la façade de L'étage
    niveau : numéro de L'étage en partant de 0 pour le Rdc
    Remarque
    Cette fonction dessine un étage d'un immeuble
    '''

    # dessin des murs

    # dessin des 3 Eléments
    pass

if __name__ == '__main__':
    etage(0,0,"red",0)
    # On ferme la fenêtre s'il y a un clic gauche
    turtle.exitonclick()
```

Les imports des modules extérieurs. Ne pas modifier !


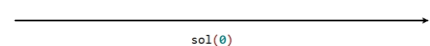


Docstring, ne Pas modifier








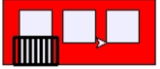
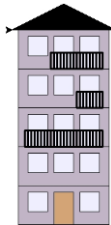
A compléter.

Le corps de la fonction

Pour le test du module, ne pas modifier !

Les fonctions à écrire sont les suivantes, par ordre de priorité d'écriture (1 en début et 6 à la fin) :

N°	P	Fonctions	
F01	1	<b>Fonction trait(x1,y1,x2,y2) :</b> Trace un trait du point de coordonnées (x1,y1) au point de coordonnées (x2,y2) x1, y1, x2, y2 : type entier	
F02	2	<b>Fonction sol(ySol) :</b> Trace un trait horizontal gras représentant le sol de la rue situé au niveau (ordonnée) ySol. ySol : type entier	
F03	2	<b>Fonction toit1(x, ySol, niveau) :</b> Trace un triangle dont le centre de la base a pour coordonnées (x,y), y dépendant du niveau du sol (ySol) et du numéro de l'étage concerné (0 étant le Rdc).	
F04	2	<b>Fonction toit2(x, ySol, niveau) :</b> Trace un trait gras dont le centre de la base a pour coordonnées (x,y) y dépendant du niveau du sol (ySol) et du numéro de l'étage concerné (0 étant le Rdc).	

F05	2	<b>Fonction rectangle(x,y,w,h):</b> Trace un rectangle dont le centre de la base a pour coordonnées (x,y), pour largeur w et pour hauteur h.	 rectangle(0,0,150,100)
F06	3	<b>Fonction porte(x,y,couleur):</b> Trace un rectangle dont le centre de la base a pour coordonnées (x,y) et une teinte donnée par le paramètre couleur.	 porte(0,0,"red")
F07	3	<b>Fonction facade(x, ySol, couleur, niveau)</b> Trace un rectangle dont le centre de la base a pour coordonnées (x,y), y étant dépendant du niveau du sol (ysol) et du numéro de l'étage concernée (0 étant le RdC) avec une teinte donnée par le paramètre couleur.	 facade(0,0,"red",0)
F08	3	<b>Fonction fenetre(x,y):</b> Trace un rectangle dont le centre de la base a pour coordonnées (x,y) de couleur bleu clair.	 fenetre(0,0)
F09	3	<b>Fonction fenetreBalcon(x,y):</b> Trace un rectangle dont le centre de la base a pour coordonnées (x,y) de couleur bleu clair.	 fenetre_balcon(0,0)
F10	3	<b>Fonction toit(x, ySol, niveau):</b> Trace un toit choisi aléatoirement entre Les deux types de toit dont le centre de la base a pour coordonnées (x,y), y dépendant du niveau du sol (ysol) et du numéro de l'étage concerné (0 étant le RdC).	(l'un ou l'autre)  toit(0,0,0)
F11	4	<b>Fonction rdc(x, ySol, cFacade, cPorte):</b> Trace un rez de chaussée dont le centre de la base a pour coordonnées (x,ysol) de couleur définie par cfacade et dont la porte a pour couleur cporte.	 rdc(0,0,"red","green")
F12	4	<b>Fonction etage(x, ySol, couleur, niveau):</b> Trace un étage dont le centre de la base a pour coordonnées (x,y), y dépendant du niveau du sol (ysol) et du numéro de l'étage concernée (0 étant le RdC) de couleur définie par couleur.	 etage(0,0,"red",0)
F13		<b>Fonction couleurAleatoire():</b> Cette fonction renvoie un P-uplet (triplet) de 3 nombres entiers compris entre 0 et 255 correspondants au codage des couleurs RVB (Rouge/Vert/Bleu)	
F14	5	<b>Fonction immeuble(x, ySol):</b> Trace un immeuble dont le centre de la base a pour coordonnées (x,ysol).	 immeuble(0,0)
F15	6	<b>Fonction principale rue():</b> Dessine la rue composée de 4 immeubles.	(Cf. exemples)

## 4 DEROULEMENT DU PROJET

**Nombre de séances envisagées :** 5 à 7 de 2 heures environ (nous adapterons selon le déroulement du projet)

Vous devez :

- **Former** un groupe de 4 élèves maximum.
- **Ecrire** la première fonction ensemble. C'est la fonction `trait()`. Toutes les autres fonctions en dépendent. Cela vous permettra en plus d'appréhender le module Turtle (cf. documents ressources).
- **Répartir** le travail (les 14 modules, fonctions à écrire). **Compléter** le fichier numériquement « 02 - [Repartition Module par Groupe.odt](#) » et **l'envoyer** à votre professeur.
- **Ecrire** les modules en commençant obligatoirement par leur ordre de priorité d'écriture.
- **Tester** le module avant de commencer un autre.

**L'évaluation :** nous attendons un compte rendu collectif sous forme numérique (se référer au fichier « 03 - [Compte-rendu.odt](#) ») relatant la description des différents modules qui ont été écrits (un par élève). Chaque membre doit y participer et décrire sa partie. Un oral sera peut-être exigé. A confirmer suite au projet.

Quelques critères qui vont nous guider pour l'évaluation :

- Le découpage de votre programme en fonction (non évalué ici, puisque le travail a été fait),
- Choix des structures dans les fonctions (le justifier si nécessaire dans le compte-rendu)
- Commentaires du code (commenté le code lorsque cela est pertinent)
- Choix du nom des variables, des fonctions (évitons d'appeler une variable par une lettre unique qui ne signifie rien, hormis x, y, et z pour des coordonnées par exemple, et pour les incréments de boucle où les i, j, k sont autorisés)
- Votre comportement pendant le projet (autonomie, investissement, le travail en équipe, rôle de coordinateur etc.)
- La conclusion en fin du compte rendu (ce que le projet vous a apporté, vos difficultés éventuelles, le travail en équipe, si vous avez une idée de projet, etc.)

**En début de séance :** faites le point afin de savoir où tous les membres de l'équipe en sont. Un responsable est désigné (un différent à chaque fois) dont le rôle est de coordonner l'équipe d'en être l'interlocuteur avec le professeur et de lui rendre compte.

**Pour les plus rapides :** Vous pouvez ajouter des éléments à votre rue : autre type de porte, des arbres, des fleurs, des grilles devant les immeubles, laisser libre cours à votre imagination... tout en respectant l'aspect modulaire du projet. Bien faire apparaître dans le compte rendu les améliorations que vous apporterez. C'est un bonus.